Improved Skeleton Extraction and Surface Generation for Sketch-based Modeling

Florian Levet* Xavier Granier†

IPARLA project (INRIA futurs - LaBRI),
UMR 5800, Universite Bordeaux 1
351, cours de la Libration
33405 Talence, France

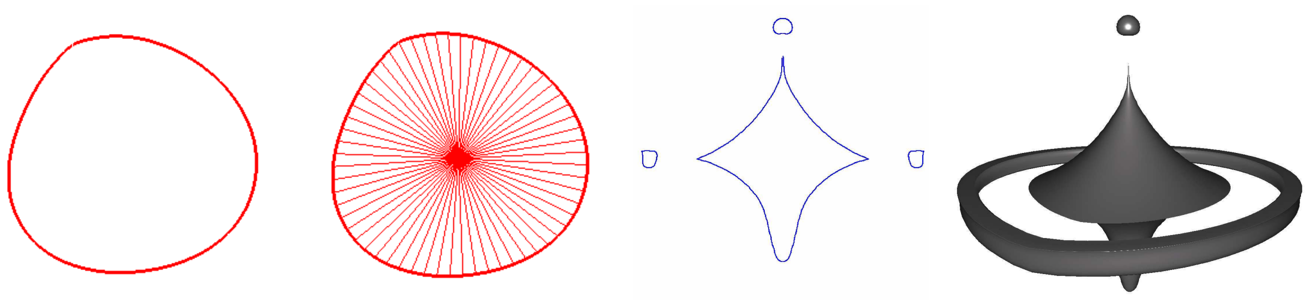


Figure 1: Surface Generation (from left to right): sketched silhouette, extracted skeleton and internal edges, sketched profile curve and resulting model.

Abstract

For the generation of freeform models, sketching interfaces have raised an increasing interest due to their intuitive approach. It is now possible to infer a 3D model directly from a sketched curved. Unfortunately, a limit of current systems is the poor quality of the skeleton automatically extracted from this silhouette, leading to low quality meshes for the resulting objects.

In this paper, we present new solutions that improve the surface generation for sketch-based modeling systems. First, we propose a new algorithm that extracts a smoother skeleton compared to previous approaches. Then, we present a new sampling scheme for the creation of good-quality 3D mesh. Finally, we propose to use a profile curve composed of disconnected components in order to create models which genus is greater than 0.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Modeling packages; I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Curve, surface, solid, and object representations;

Keywords: Modeling, FreeForm Surfaces, Sketching, Implicit Surface

1 Introduction

Due to the large number of tools, the interface of current 3D modelers are extremely complex and intimidating for non-expert users. Creating a prototype of a 3D model can thus become a slow and painful task. So, in order to provide simpler interfaces, new approaches for 3D modeling have been developed, based on the human ability to quickly draw a global overview of an object. These approaches are commonly referred as *3D Sketching*. Their principle is to infer the shape of a 3D model based on sketched 2D curves. On this model, users can add details thanks to different editing operations (cutting, extrusion, etc,.), all based on sketching interaction. Since all these systems use a drawing metaphor, they can be easily adopted even by non-expert users.

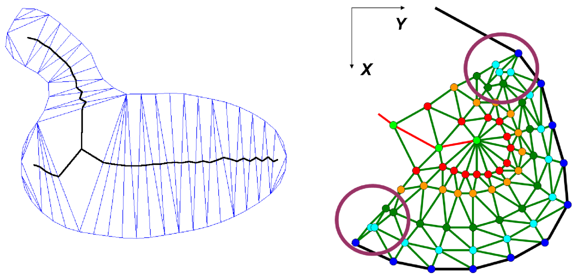
From the first freeform modeler based on sketching [10], many works have been published that try to improve the surface generation [11, 4, 8, 20, 19, 7] and the editing operations [18, 20, 7, 14]. Among all the editing operations, changing the profile curve [20, 7, 14] allows an efficient modification of the local or global shape, and the generation of more complex models in few sketches.

In this paper, we introduce three main contributions. (i) The first one is a new solution for the skeleton extraction from a silhouette curve. This skeleton, computed thanks to a 2D variational implicit surface [22] that approximates the silhouette curve, is smoother than previous algorithms (see Section 4). (ii) The second contribution is a new approach for inferring the 3D volume based on this skeleton. Its main goal is to provide a better quality mesh without inherent artefacts that occur on mesh-based freefrom modeling (see Section 5). (iii) Finally, we propose a solution to generate a model with genus greater than 0 based on more complex profile curves.

2 Previous work

Teddy [10] is the precursor system for the creation of 3D freeform models by using gestures and curves. In order to infer a 3D model from a silhouette, this curve is sampled and the resulting samples are triangulated with a Constrained Delaunay Triangulation. Based on this triangulation, a skeleton is extracted. The elevation is computed on each vertex of this skeleton based on the average distance to the nearest silhouette samples. This skeleton extraction heavily depends on the initial triangulation, and is prone to several artefacts (see Figure 2), leading to a low quality mesh [9]. In order

*levet@labri.fr
†granier@labri.fr



(a) (b)

Figure 2: Limitations of the Teddy approach: (a) the skeleton extracted from the CDT (Constrained Delaunay Traingulation) has some discontinuities and, (b) the sampling of the internal edges leads to very tiny and uneven triangles.

to improve the quality of the skeleton, some vertices can be removed [20], but the resulting curve still depend on the triangulation and thus on the curve sampling process.

Despite these limitations, the algorithm developped in Teddy is used in many related publications [18, 4, 8, 24, 20, 3, 14], mainly due to its simplicity and its versatility. As an example, Owada et al. [18] extend it to voxel-based volume generation. Some other works [4, 8, 20, 24, 3] used variational implicit surfaces in order to overcome the problem of mesh quality. By using a simplified skeleton combined with the blending properties of the implicit surfaces, they were able to improved the model quality. Unfortunately, global editing of the geometry is very expensive since the whole implicit model needs to be reconstructed and repolygonized before visualization.

Among all the previous solutions, only profile-based ones [20, 7, 14] enable the creation of models that are not limited to blobby shapes. Nevertheless, all of them have some specific limitations and strongly rely on the skeleton quality. Because of the underlying surface representation (a polynomial-weighted convolution surface), the definition of the profile curve in the *ConvMo* [20] environment is limited to polar coordinates. The segment-based skeleton is improved by removing some vertices, but it still depends on the initial sampling of the silhouette. In our previous work [14], we have presented a system that can create genus 1 models. But the resulting mesh relies even more on the initial quality of the skeleton. Cherlin et al. [7] do not use a skeleton, but an interpolating parametric surface in order to create models that are not limited to blobby objects. Unfortunately, because they do not extract a skeleton, the resulting models are limited to a grid topology and thus can not have junctions.

Alexe et al. [2] used an image processing technique for the creation of the skeleton. Thus, from the silhouette curve and a distance image, they can extract a skeleton that do not depend on the initial sampling. This skeleton is composed of segments and polygons. The reconstructed surface is thus flatter in regions where the skeleton is a polygon. However, this algorithm is not well adapted to profile edition that requires curve or segment-based skeletons.

Numerous methods have been proposed to extract the skeleton of a 2D shape. They can be classified in two main families: *discrete methods* and *continuous methods*. Continuous methods are generally based on the Voronoi graph of a point set located on the object boundary [5, 6, 17]. The dual of the Voronoi diagram, the Delaunay triangulation has also been used extensively [1, 23] as in Teddy [10]. Both approaches preserve topology but, unfortunately, the techniques used to prune faces and edges which correspond to small perturbations of the boundary are mostly based on heuristics.

The discrete methods work directly on binary images. For instance, methods based on *thinning* [13, 15, 12, 16] attempt to create a skeleton which preserves the shape topology thanks to the definition of some erosion rules. However, these methods are quite sensitive to Euclidean transformations and, besides, they usually yield un-wanted edges.

Some sketching systems [19, 7] do not use a skeleton to inflate the object volume. Karpenko et al. [11] inflate the object volume by elevating the silhouette gravity center. Some problems may occur when the gravity center is not inside the object silhouette. Schmidt et al [19] used a 2D variational implicit surface as an approximation of the silhouette curve. Then, this 2D field is swept along an infinite 3D axis and bounded. Thanks to this technique, they can create blobby objects, sweep surfaces and surfaces of revolution.

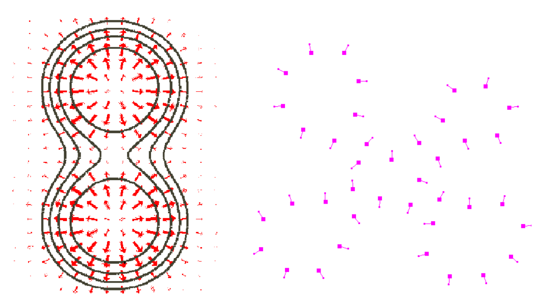
3 Motivation

The more versatile solution to generate a 3D surface from a 2D silhouette curve still seems to be the algorithm developed in Teddy. But, as said before, this algorithm has some strong limitations. First, the use of a Constrained Delaunay Triangulation (CDT) computed from the silhouette curve, leads to the creation of a skeleton with a lot of local oscillations as can be seen in Figure 2(a), even in very simple cases. This results in a 3D object with a lot of height variations along the skeleton. These variations are propagated on the final mesh. While they are less visible with a classical circular profile curve, these artefacts are increased with a freeform profile curve. Besides, even with an improved skeleton [20], different samplings of the silhouette lead to different triangulations and thus, to different skeletons and resulting meshes.

The second problem occurs for the mesh-based solution and depends on the sampling along the internal edges (an internal edge is an edge that connects a silhouette point and a skeleton point). Once elevated, these samples will result in the final vertices of the mesh. Since these internal edges are based on the triangulation, more than one internal edge can end on one silhouette point (see Figure 2(a)). As can be seen in Figure 2(b), this leads to the creation of very tiny and uneven triangles near these silhouette samples. The low quality of the mesh can create some artefacts during the editing operations.

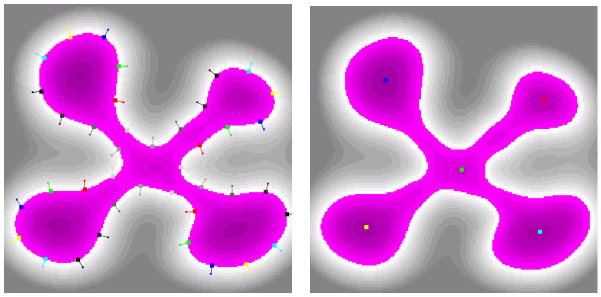
In order to improve the quality of the inferred mesh, we have identified two desirable properties for the skeleton and the internal edges:

1. the skeleton should be smooth to limit the local variations,
2. each silhouette point should be connected to only one internal edge.



(a) (b)

Figure 3: (a) Gradient field of an implicit surface (from [26]). (b) Silhouette points used to reconstruct the 2D variational implicit surface.



(a) (b)

Figure 4: (a) Variation of the reconstructed implicit function with some silhouette samples. (b) These samples have moved according to the gradient field and converged to the vanishing points, corresponding to the seeds of our skeleton.

To fulfil these properties, we present a two-fold alternative technique compared to existing works. First, in Section 4, we propose to compute a smooth skeleton by using a 2D variational implicit surface. Then, in Section 5, we present a new sampling technique which ensures that only one internal edge corresponds to a silhouette point.

4 Creation of the skeleton

Similarly to the work of Zonenschein et al. [25] on texturing of implicit surfaces, the basic idea of our technique is to use the gradient of a 2D variational implicit surface as a force field (see Figure 3(a)). Thanks to this force field, the seed points (we will simply call them seeds) of the skeleton are computed using a two-steps process.

4.1 Finding the skeleton seeds

First, a 2D variational implicit surface [22] is used as an approximation of the silhouette curve sketched by the user. As can be seen in Figure 4, the implicit function has some local minima. The corresponding points are called vanishing points (points where the gradient of the implicit function is null) and they represent the seeds of our skeleton.

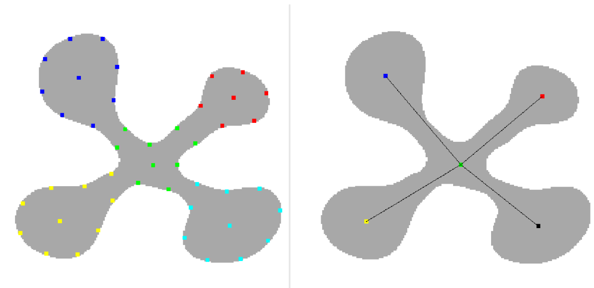
In order to find the position of the seeds, the gradient of the implicit surface is used as a force field. A gradient descent is applied on some silhouette samples (see Figure 4(a)) until their convergence to the vanishing points (as shown in Figure 4(b)). In the Figure 4, the silhouette has four branches, resulting in five skeleton seeds. This provides us a good representation of the silhouette topology.

4.2 Neighboring relations of the seeds

In order to create the full skeleton, the seeds have to be connected. For this purpose, we compute their neighboring relationship. Thanks to the gradient descent, each seed can be associated with the set of its original silhouette samples (samples that have converged to this seed). If two adjacent silhouette samples do not converge to the same seed, the two corresponding seeds are neighbors. The neighboring relations between the seeds computed in Figure 5 correspond to the four branches of the original curve.

4.3 Creation of new skeleton points

In order to create the full skeleton, the curve connecting two neighbor seeds has to follow the line of local minimum of the implicit function. An approximation of this curve will be computed by



(a) (b)

Figure 5: (a) Gathering behavior of the silhouette samples: their color is the same that their corresponding seed. (b) Neighboring relations between the seeds.

a uniform sampling in-between two connected seeds. The distance between two consecutive skeleton points is evaluated to be $d = |r_{ij}|/\Delta$, where r_{ij} is the vector between the two neighboring seeds i and j and Δ the desired number of points.

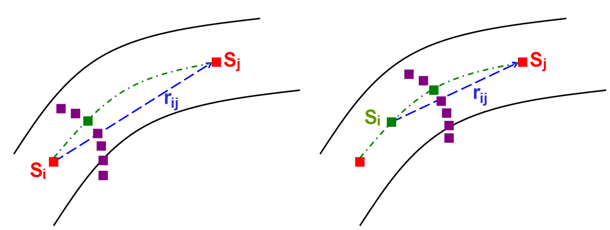
For a faster creation of these skeleton points, the process is only based on the evaluation of the implicit function, and the search of a local minimum. In order to find this minimum, a set of sample points on a circular section are computed in the direction of the next seed, with d as radius and the current seed as its center (Figure 6(a)). On the circular section, the position with the minimal value of the implicit function is added as a new point. This process is repeated on the newly added point (Figure 6(b)) until the distance between the newly inserted point and the second seed is lower than the required distance. This sampling is repeated for all the neighboring relations of the seeds. Since the resulting points follow the local minimum of the implicit function, the skeleton inherits its smoothness from the implicit function (as shown in Figure 16).

5 Inferring the volume of the object

Now that a smooth skeleton has been extracted from the silhouette curve, the next step of the process is to infer the volume of the object. For a final higher quality mesh, our system introduces a new creation and sampling of internal edges.

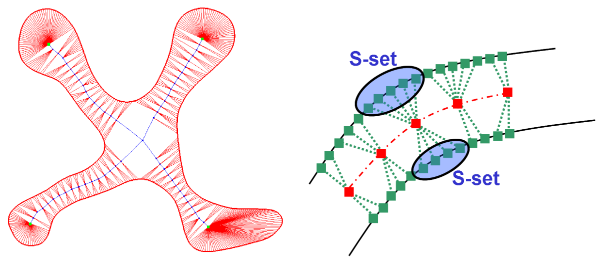
5.1 Relations between silhouette samples and skeleton points

The skeleton seeds (see Section 4) are computed from a set of silhouette samples. These samples are in fact a small subset of the



(a) (b)

Figure 6: Creation of new skeleton points. The local minimum of the implicit function is the green dotted line; green points are the added skeleton points while violet points are the non-local minimums of the implicit function evaluation. (a) Starting from the first seed, a new skeleton point is added (corresponding to a local minimum). (b) The new point becomes the new starting point.



(a) (b)

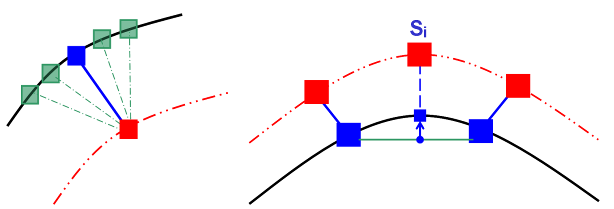
Figure 7: (a) Connecting relations between skeleton points and silhouette samples. (b) Determination of the S-set of a given skeleton point.

complete sampling of the profile curve, since for finding the vanishing points, only a coarse approximation of the curve is required (in theory, a minimum of one silhouette sample per seed). On the contrary, the creation of the internal edges is facilitated if there are more silhouette samples than skeleton points. Discarding useless samples is easier than adding new points in under-sampled region of the silhouette curve. For the following algorithm, the whole set of silhouette samples is used.

The first step of the process is to gather, for each skeleton point, the closest silhouette samples. Based on these connecting sets, the final internal edges are computed based on the following three properties:

- There is one internal edge at most for each silhouette sample. This property removes the meshing problem shown in Figure 2(b).
- From each extremal skeleton point (point connected to only one neighbor), internal edges are built to link all the silhouette samples which have this point as closer skeleton point (see the green points in Figure 7(a)). The resulting triangle fans accurately represent extremal sections of the silhouette.
- From the other skeleton points connected to n other neighboring points ($n > 1$), n internal edges are issued to n silhouette samples that have to be non-adjacent (cf. Figure 7(a)).

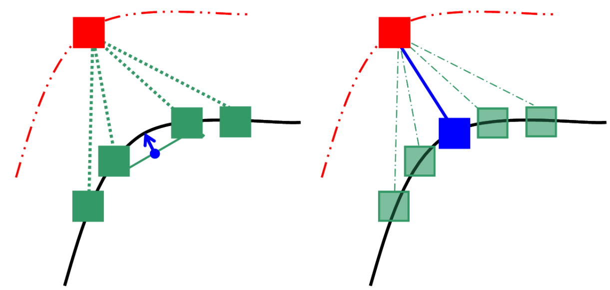
Based on the connecting sets and these defined properties, the internal edges can be created. For the extremal points, a subset of connected silhouette samples is selected in order to create the triangle fan. For all the other skeleton points, the process is the same. Depending on the number n of neighbor points, its connect silhouette samples are divided in n non-adjacent sets that we call *S-sets* ($n = 2$ in Figure 7(b)).



(a) (b)

Figure 8: (a) Creation of an internal edge for an odd number of samples in the S-set. The blue sample is chosen while the green samples are discarded. (b) If an S-set is empty, a new sample is added between the two silhouette samples of the two closest internal edges. Then, an internal edge is created between this sample and S_i.

5.2 Creation of the internal edges



(a) (b)

Figure 9: Creation of an internal edge for an even number of samples in the S-set. (a) A new sample is created between the two middle samples of the S-set. (b) This sample is projected on the surface and an internal edge is created between this sample and the skeleton point.

We can see the S-sets of a skeleton point as the different sections of silhouette that are controlled by its position. For the creation of an internal edge, the best sample of the S-set is thus the middle of each section.

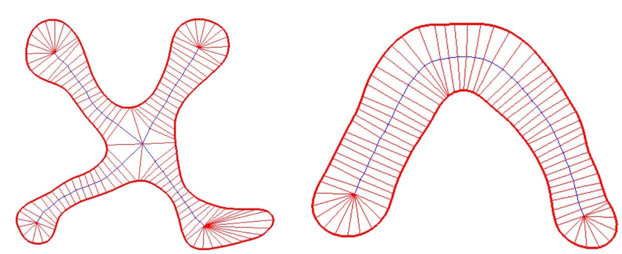
If the number of samples of an S-set is odd, an internal edge is created between the middle sample and the skeleton point (see Figure 8(a)) and all the other S-set samples are then discarded. On the contrary, if the number is even, a new silhouette sample is added between the two middle samples and projected onto the implicit surface (see Figure 9(a)). An internal edge is created with the skeleton point and all the other S-set samples are discarded (see Figure 9(b)).

At the end of the algorithm, some internal edges can be missing (see Figure 8(b)). This corresponds to empty S-sets due to undersampled region. In order to create a missing internal edge, the system gathers the two closer internal edges and creates a new sample on the curve between their two corresponding silhouette samples. This new sample is then connected to the current skeleton point.

The result of this complete process is a set of internal edges that correspond to the desirable properties defined in Section 5.1 (see Figure 10) and, thus, that will lead to good quality 3D meshes.

5.3 Creation of genus 1+ models with profile curve

Using profile curves in order to edit locally or globally the freefrom shape [20, 7, 14] enables more complex models in a few strokes, but this extension to non-convex curves needs a more precise skeleton, as noted in [20]. We can use such an approach with our smooth skeleton and our internal edges extraction: the results are improved, as shown in the comparison between our system (Figure 13(a-c))



(a) (b)

Figure 10: Examples of the internal edges creation for two different silhouette curves.

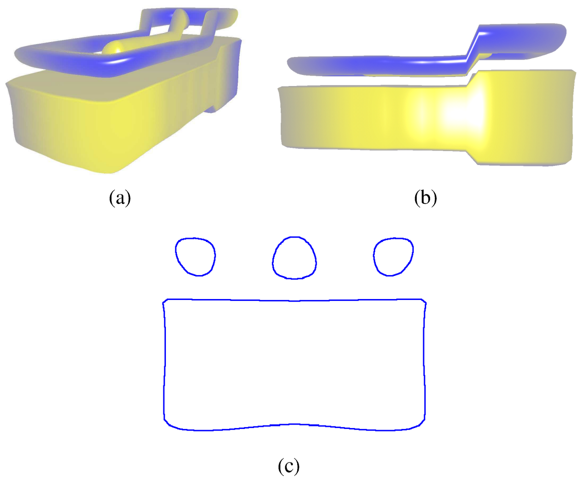


Figure 11: Creation of a genus 1+ model. (c) The profile curve is composed of three disconnected components. (a)-(b) The resulting 3D object.

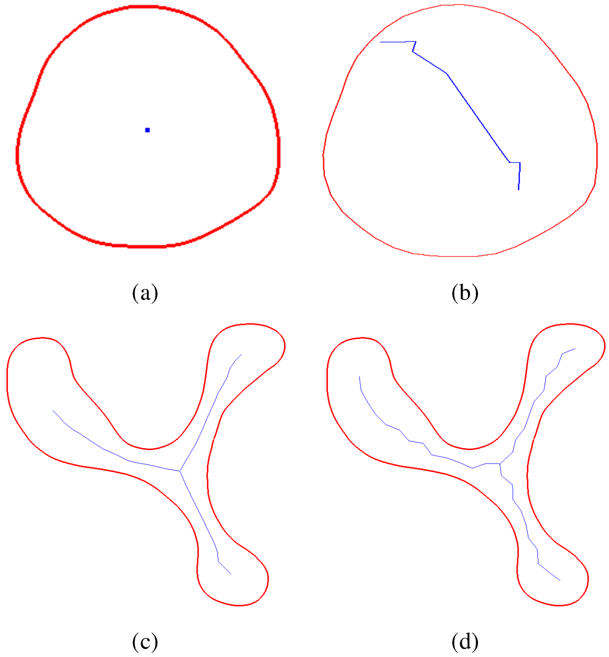


Figure 12: Comparison between the skeleton of Teddy (b)-(d) and ours (a)-(c). Thanks to implicit curve our algorithm manages to create a good skeleton (one point) for circular silhouette curves. Besides, for more complex silhouette curves, the skeleton extracted by our technique is smoother.

and our previous work [14] (Figure 13(b-d)). For this model, we have sketched rectangular silhouette and profile curves.

Our previous system [14] is limited to genus 1 models. We introduce here how to generate models with larger genus (genus > 1). The user is now not limited to only one profile curve. He can sketch several components (as shown Figure 11(c)) before the profile modification is applied on the 3D model. Then, for each internal edge, each disconnected component of the profile curve is sampled, and these samples are reprojected along each internal edge (for a more precise explanation on how the resulting mesh is reconstructed with respect to the profile curve sketched by the user please refer to [14]). The different components appear in the final model, resulting in genus 1+ solutions, as shown in Figure 11.

6 Results and discussion

One of the main advantages of our approach is a higher quality of the resulting skeleton. Figure 12 shows a comparison between our skeleton extraction and the one from Teddy [10]. As explained in Section 4, the use of an implicit function leads to the reconstruction of a smoother skeleton without the un-wanted local oscillations (see Figure 12(a-c)). For instance, the Teddy algorithm always extracts a skeleton composed of connected segments even for a near-circular silhouette curves (see Figure 13(b)). The final model will thus not be similar to a sphere. Our algorithm extracts a skeleton composed of a unique point (see Figure 12(b) and Figure 1). The resulting internal edges have a much better quality (see Figure 1) and the inflated 3D model will have less rendering artefacts. Additional skeletons extracted using our technique can be seen in Figure 16. Note that each skeleton is a good approximation of the silhouette topology.

As said previously, a smooth skeleton is needed when using profile editing operations. Indeed, skeleton discontinuities lead to height differences in the vertices of the final mesh. Figure 13 shows some comparisons between our system (Figure 13(a-c)) and our previous one [14] one (Figure 13(b-d)) for two different models. For each model, the same silhouette and profile curves were used. Because we have a smoother skeleton and a better sampling, the models reconstructed with our system are smoother and do not present the height differences on the borders of the models.

By sketching profile curves with disconnected components, our system allows the users to generate models with genus greater than 1 (see Figure 1 and Figure 11). For example, we have created the *GI* logo (see Figure 14). It was modeled with 3 different objects (the profile curves used for each object are shown) within a total of six curves (2 for each object). Because of the use of a profile with disconnected components and internal holes, the genus of the letter *G* is 4 (please note that no parametrization was needed, the user only sketched the silhouette of the letter *G* and the two components of the corresponding profile curve). Besides, unlike most existing systems, our system creates good-quality models when used with non-circular profile curves (see Figure 1, 11 and 14).

The main limitation of our current technique is the determination of the vanishing points. Indeed, they are only dependant of the definition of the implicit function. For instance, if the user draws a silhouette curve with a large enough zig-zag then some vanishing points will correspond to this zig-zag. The problem is that we can not control the apparition of these vanishing points. Besides, in some cases, the user may want to have vanishing points corresponding to this zig-zag, but in other cases, he may not want to. So, we think that the best solution is to have a semi-automatic generation of the skeleton, the user having the possibility to edit the vanishing points (by discarding some vanishing points, adding news points or moving the existing ones).

7 Conclusion

In this paper, we have presented new techniques in order to improve the quality of the models generated using sketch-based freeform modeling systems. The silhouette curve drawn by the user is first approximated by a 2D variational implicit function. From this variational approximation, a skeleton is extracted that inherits from the implicit function its global smoothness. We have also introduced a new scheme to create the set of internal edges which aim is to limit the creation of tiny and uneven triangles. Indeed, better internal

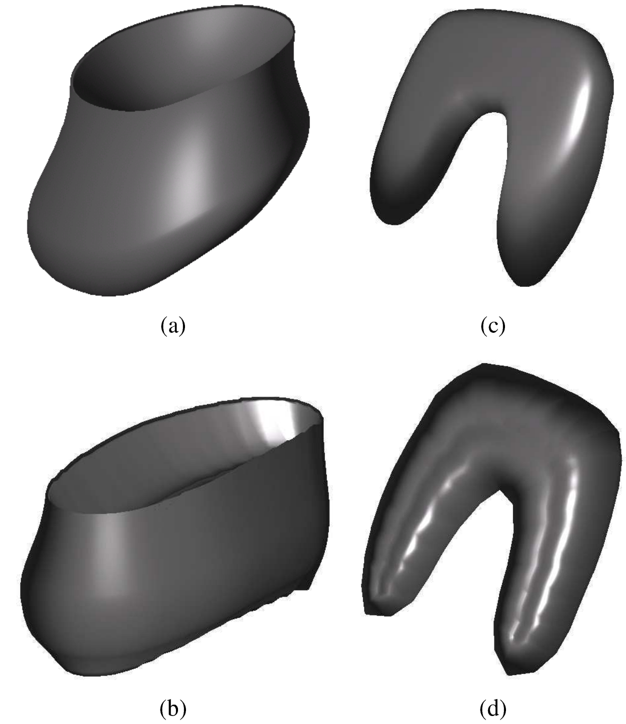


Figure 13: Comparison between two models generated by our current system (a)-(c) and our previous one (b)-(d). For each object, the same silhouette and profile curves were used. Because of our smoother skeleton, models generated with our system do not have the classical rendering artefacts generated by a skeleton with discontinuities.

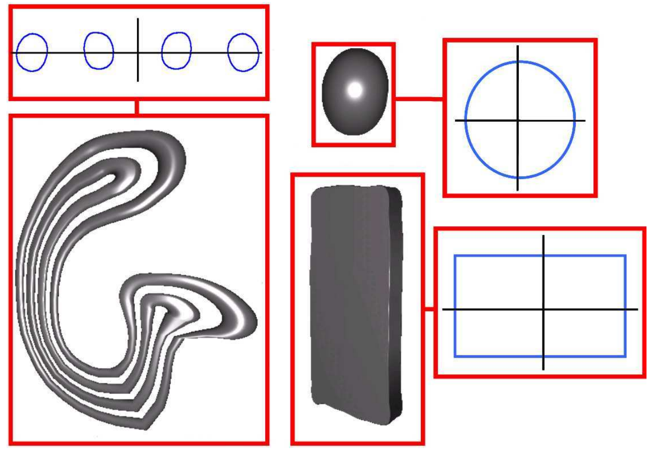


Figure 14: The 3D *GI* logo was created with three different objects. The profile curves applied on each object are shown. The letter *G* is a model which genus is > 1 since its profile curve has disconnected components.

edges lead to the creation of better meshes for the final object. Finally, we have presented sampling process in order to create models with genus greater than 0, based on the sketch of profile curves with holes and disconnected components.

After focusing on the improvement of the surface construction, we are now investigating new interactions for sketch-based modeling systems. Borrowing some elements from [21], our current direction is the integration of different drawing plans on the object view. We believe that offering the possibility to modify different views of the same object can lead to the construction of complex and interesting models. One other direction, is to develop different editing operations based on the properties of our reconstructed mesh.

Acknowledgments

We wish to thank Christophe Schlick and the anonymous reviewers for their helpful remarks and corrections.

References

[1] M. Af, O. Ubler, R. Kikinis, M. Shenton, and G. Ekely. Characterization and recognition of 2d organ shape in medical image analysis using skeletonization. In *In Proc. Workshop on Mathematical Methods in Biomedical Image Analysis (MMBIA)*, pages 139–150, 1996.

[2] A. Alexe, L. Barthe, M.P. Cani, and V. Gaildrat. Shape modelling by sketching using convolution surfaces. In *Pacific Graphics (Short Papers)*, October 2005.

[3] A. Alexe, V. Gaildrat, and L. Barthe. Interactive modelling from sketches using spherical implicit functions. In *Proc. AFRIGRAPH '04*, pages 25–34. ACM Press, 2004.

[4] B. De Araujo and J. Jorge. Blobmaker: Free-form modelling with variational implicit surfaces. In *Proc. of "12 Encontro Portugus de Computao Grafica" (12 EPCG)*, pages 17–26, October 2003.

[5] D. Attali and A. Montanvert. Semicontinuous skeletons of 2d and 3d shapes. In *Proc. of the 2nd International Workshop on Visual Form*, pages 32–41, 1994.

[6] J. W. Brandt and R. Algazi. Continuous skeleton computation by Voronoi diagram. *CVGIP: Image Underst.*, 55(3):329–338, 1992.

[7] J. J. Cherlin, F. Samavati, M. C. Sousa, and J. A. Jorge. Sketch-based modeling with few strokes. In *SCCG '05: Proceedings of the 21st spring conference on Computer graphics*, pages 137–145. ACM Press, 2005.

[8] A. Cuno, C. Esperana, P. Roma, and R. Farias. 3D Free-Form Modeling with Variational Surfaces. In *WSCG '05*, 2005.

[9] T. Igarashi and J. F. Hughes. Smooth meshes for sketch-based freeform modeling. In *SI3D '03: Proceedings of the 2003 symposium on Interactive 3D graphics*, pages 139–142. ACM Press, 2003.

[10] T. Igarashi, S. Matsuoka, and H. Tanaka. Teddy: a sketching interface for 3d freeform design. In *Proc. ACM SIGGRAPH '99*, pages 409–416, 1999.

[11] O. Karpenko, J. Hughes, and R. Raskar. Free-form Sketching with Variational Implicit Surfaces. *Computer Graphics Forum (Proc. Annual Eurographics Conference 2002)*, 21(3):585–594, 2002.

[12] L. Lam, S.-W. Lee, and C. Y. Suen. Thinning Methodologies-A Comprehensive Survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(9):869–885, 1992.

[13] Ta-Chih Lee, Rangasami L. Kashyap, and Chong-Nam Chu. Building skeleton models via 3-d medial surface/axis thinning algorithms. *CVGIP: Graph. Models Image Process.*, 56(6):462–478, 1994.

[14] Florian Levet, Xavier Granier, and Christophe Schlick. 3d sketching with profile curves. In *International Symposium on Smart Graphics*, 2006.

[15] Antoine Manzanera, Thierry M. Bernard, Francoise J. Preteux, and Bernard Longuet. Medial faces from a concise 3d thinning algorithm. In *ICCV (1)*, pages 337–343, 1999.

[16] Virginie Marion-Poty and Serge Miguet. A new 2-d and 3-d thinning algorithm based on successive border generations. In *4th Discrete Geometry for Computer Imagery*, pages 195–206, 1194.

[17] R. L. Ogniewicz and O. Kübler. Hierarchic Voronoi skeletons. *Pattern Recognition*, 28(3):343–359, 1995.

[18] S. Owada, F. Nielsen, K. Nakazawa, and T. Igarashi. A Sketching Interface for Modeling the Internal Structures of 3D Shapes. In *Proc. Smart Graphics*, pages 49–57. Springer-Verlag, July 2003.

[19] R. Schmidt, B. Wyvill, M.C. Sousa, and J.A. Jorge. ShapeShop:

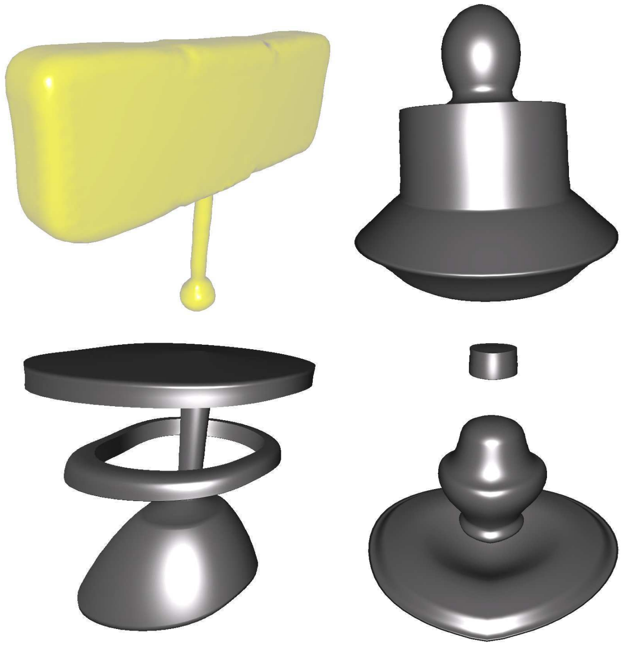


Figure 15: Different objects created with our system.

Sketch-Based Solid Modeling with BlobTrees. In *2nd Eurographics Workshop on Sketch-Based Interfaces and Modeling*, pages 53–62, 2005.

[20] C.-L. Tai, H. Zhang, and C.-K. Fong. Prototype Modeling from Sketched Silhouettes based on Convolution Surfaces. *Computer Graphics Forum*, 23(1):71–83, 2004.

[21] S. Tsang, R. Balakrishnan, K. Singh, and A. Ranjan. A suggestive interface for image guided 3d sketching. In *CHI '04: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 591–598. ACM Press, 2004.

[22] G. Turk and J. O'Brien. Variational implicit surfaces. Technical Report GIT-GVU-99-15, Georgia Institute of Technology, 1998.

[23] X. Yu, J. A. Goldak, and L. Dong. Constructing 3-D discrete medial axis. In *SMA '91: Proceedings of the first ACM symposium on Solid modeling foundations and CAD/CAM applications*, pages 481–489. ACM Press, 1991.

[24] R. Zenka and P. Slavik. New dimension for sketches. In *SCCG '03: Proceedings of the 19th spring conference on Computer graphics*, pages 157–163. ACM Press, 2003.

[25] R. Zonenschein, J. Gomes, L. Velho, and L. H. de Figueiredo. Texturing implicit surfaces with particle systems. In *ACM SIGGRAPH 97 Visual Proceedings*, 1997.

[26] R. Zonenschein, J. Gomes, L. Velho, L. H. de Figueiredo, M. Tigges, and B. Wyvill. Texturing Composite Deformable Implicit Objects. In *SIBGRAPI'98*, pages 346–353, 1998.

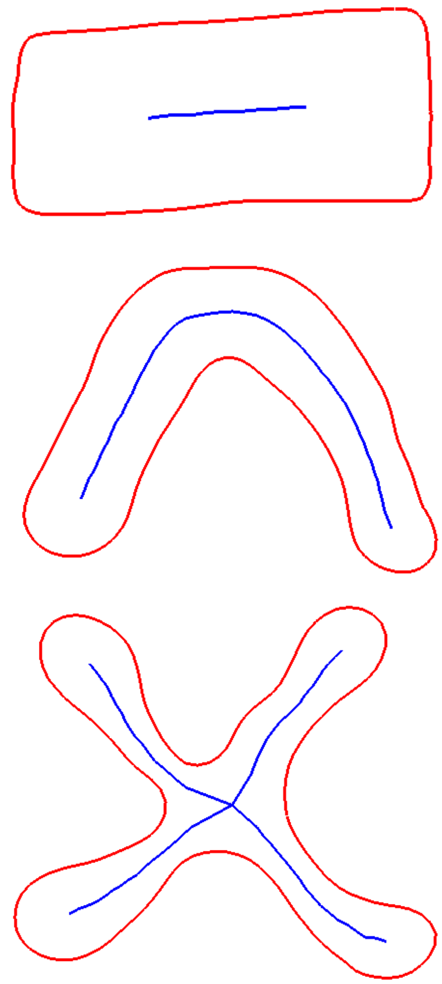


Figure 16: Skeletons of different silhouette curves.